

Amendments to the Claims

1 Claim 1 (previously presented): A computer program product for providing fast and efficient
2 address lookup for an address comprised of a plurality of address components and wherein each
3 address component is deemed to be more significant than its next-sequential neighboring address
4 component, the computer program product embodied on one or more computer-readable media
5 and comprising:

6 computer-readable program code means for creating a plurality of arrays comprising an
7 array for each of the address components, wherein each array comprises a plurality of entries
8 which are indexed using values of the address component for which the array was created, further
9 comprising:

10 computer-readable program code means for obtaining a particular address value to
11 be represented in the plurality of arrays;

12 computer-readable program code means for obtaining a bit mask associated with
13 the particular address value;

14 computer-readable program code means for indexing into a highest-order one of
15 the arrays using a most-significant component of the particular address value as an index
16 element;

17 computer-readable program code means for setting a flag associated with the
18 index element to on if the bit mask indicates that the next-sequential neighboring address
19 component is considered significant, and for setting the flag to off otherwise; and

20 computer-readable program code means for repeating the indexing and setting,
21 using the next-highest-order one of the arrays and the next-most-significant component of the

particular address value, while the bit mask indicates that the next-sequential neighboring address component is considered significant, and for (1) storing information associated with the particular address value in a storage or memory location and (2) setting a pointer field associated with the index element to point to the storage or memory location, otherwise; and

computer-readable program code means for retrieving the stored information associated with a selected address value from the plurality of arrays, further comprising:

computer-readable program code means for obtaining the selected address value;

computer-readable program code means for obtaining a selected bit mask associated with the selected address value;

computer-readable program code means for indexing into the highest-order one of the arrays using the most-significant component of the selected address value as the index element; and

computer-readable program code means for determining that no result is available if the index element has no stored information, and for continuing otherwise, wherein the continuing further comprises:

computer-readable program code means for checking the flag associated with the index element; and

computer-readable program code means for returning the stored information from the storage or memory location pointed to by the pointer field when the flag is set off or for repeating the indexing and determining, for the next-highest-order one of the arrays and the next-most-significant component of the selected address value, when the flag is set on.

1 Claim 2 (original): The computer program product according to Claim 1, wherein the computer-
2 readable program code means for repeating further comprises computer-readable program code
3 means for setting a use count associated with the storage or memory location to a number which
4 represents a count of the array entries which point to this storage or memory location when the
5 next-sequential neighboring address component is not considered significant.

1 Claim 3 (original): The computer program product according to Claim 2, wherein the stored
2 information in the memory or storage location comprises an associated bit mask and wherein the
3 computer-readable program code means for retrieving further comprises computer-readable
4 program code means for resolving a collision, further comprising:

5 computer-readable program code means for comparing the selected address value to each
6 bit mask associated with the stored information from multiple storage or memory locations,
7 yielding a plurality of bit mask results; and

8 computer-readable program code means for selecting a collision result using that one of
9 the bit mask results which both (1) matches the selected address value according to the selected
10 bit mask and (2) has the longest associated bit mask.

1 Claim 4 (original): The computer program product according to Claim 1, wherein the address is
2 an Internet Protocol (IP) address.

1 Claim 5 (original): The computer program product according to Claim 4, wherein the IP address
2 is an IP version 4 address and wherein there are 4 components in each IP version 4 address and

3 thus 4 arrays.

1 Claim 6 (original): The computer program product according to Claim 4, wherein the IP address
2 is an IP version 6 address and wherein there are 16 address components in each IP version 6
3 address and thus 16 arrays.

1 Claim 7 (previously presented): A system for providing fast and efficient address lookup for an
2 address comprised of a plurality of address components and wherein each address component is
3 deemed to be more significant than its next-sequential neighboring address component, the
4 system comprising:

5 means for creating a plurality of arrays comprising an array for each of the address
6 components, wherein each array comprises a plurality of entries which are indexed using values
7 of the address component for which the array was created, further comprising:

8 means for obtaining a particular address value to be represented in the plurality of
9 arrays;

10 means for obtaining a bit mask associated with the particular address value;

11 means for indexing into a highest-order one of the arrays using a most-significant
12 component of the particular address value as an index element;

13 means for setting a flag associated with the index element to on if the bit mask
14 indicates that the next-sequential neighboring address component is considered significant, and
15 for setting the flag to off otherwise; and

16 means for repeating the indexing and setting, using the next-highest-order one of

the arrays and the next-most-significant component of the particular address value, while the bit mask indicates that the next-sequential neighboring address component is considered significant, and for (1) storing information associated with the particular address value in a storage or memory location and (2) setting a pointer field associated with the index element to point to the storage or memory location, otherwise; and

means for retrieving the stored information associated with a selected address value from the plurality of arrays, further comprising:

means for obtaining the selected address value;

means for obtaining a selected bit mask associated with the selected address value;

means for indexing into the highest-order one of the arrays using the most-significant component of the selected address value as the index element; and

means for determining that no result is available if the index element has no stored information, and for continuing otherwise, wherein the continuing further comprises:

means for checking the flag associated with the index element; and

means for returning the stored information from the storage or memory location pointed to by the pointer field when the flag is set off or for repeating the indexing and determining, for the next-highest-order one of the arrays and the next-most-significant component of the selected address value, when the flag is set on.

Claim 8 (original): The system according to Claim 7, wherein the means for repeating further comprises means for setting a use count associated with the storage or memory location to a

number which represents a count of the array entries which point to this storage or memory location when the next-sequential neighboring address component is not considered significant.

Claim 9 (original): The system according to Claim 8, wherein the stored information in the memory or storage location comprises an associated bit mask and wherein the means for retrieving further comprises means for resolving a collision, further comprising:

means for comparing the selected address value to each bit mask associated with the stored information from multiple storage or memory locations, yielding a plurality of bit mask results; and

means for selecting a collision result using that one of the bit mask results which both (1) matches the selected address value according to the selected bit mask and (2) has the longest associated bit mask.

Claim 10 (original): The system according to Claim 7, wherein the address is an Internet Protocol (IP) address.

Claim 11 (original): The system according to Claim 10, wherein the IP address is an IP version 4 address and wherein there are 4 components in each IP version 4 address and thus 4 arrays.

Claim 12 (original): The system according to Claim 10, wherein the IP address is an IP version 6 address and wherein there are 16 address components in each IP version 6 address and thus 16 arrays.

Serial No. 09/680,791

-7-

Docket RSW9-2000-0050-US1

1 Claim 13 (previously presented): A method for providing fast and efficient address lookup for an
2 address comprised of a plurality of address components and wherein each address component is
3 deemed to be more significant than its next-sequential neighboring address component, the
4 method comprising the steps of:

5 creating a plurality of arrays comprising an array for each of the address components,
6 wherein each array comprises a plurality of entries which are indexed using values of the address
7 component for which the array was created, further comprising the steps of:

8 obtaining a particular address value to be represented in the plurality of arrays;

9 obtaining a bit mask associated with the particular address value;

10 indexing into a highest-order one of the arrays using a most-significant component
11 of the particular address value as an index element;

12 setting a flag associated with the index element to on if the bit mask indicates that
13 the next-sequential neighboring address component is considered significant, and setting the flag
14 to off otherwise; and

15 repeating the indexing and setting, using the next-highest-order one of the arrays
16 and the next-most-significant component of the particular address value, while the bit mask
17 indicates that the next-sequential neighboring address component is considered significant, and
18 (1) storing information associated with the particular address value in a storage or memory
19 location and (2) setting a pointer field associated with the index element to point to the storage or
20 memory location, otherwise; and

21 retrieving the stored information associated with a selected address value from the

22 plurality of arrays, further comprising the steps of:
23 obtaining the selected address value;
24 obtaining a selected bit mask associated with the selected address value;
25 indexing into the highest-order one of the arrays using the most-significant
26 component of the selected address value as the index element; and
27 determining that no result is available if the index element has no stored
28 information, and continuing otherwise, wherein the continuing further comprises the steps of:
29 checking the flag associated with the index element; and
30 returning the stored information from the storage or memory location
31 pointed to by the pointer field when the flag is set off or repeating the indexing and determining,
32 for the next-highest-order one of the arrays and the next-most-significant component of the
33 selected address value, when the flag is set on.

1 Claim 14 (original): The method according to Claim 13, wherein the repeating step further
2 comprises the step of setting a use count associated with the storage or memory location to a
3 number which represents a count of the array entries which point to this storage or memory
4 location when the next-sequential neighboring address component is not considered significant.

1 Claim 15 (original): The method according to Claim 14, wherein the stored information in the
2 memory or storage location comprises an associated bit mask and wherein the retrieving step
3 further comprises resolving a collision, further comprising the steps of:
4 comparing the selected address value to each bit mask associated with the stored

5 information from multiple storage or memory locations, yielding a plurality of bit mask results;
6 and
7 selecting a collision result using that one of the bit mask results which both (1) matches
8 the selected address value according to the selected bit mask and (2) has the longest associated
9 bit mask.

1 Claim 16 (original): The method according to Claim 13, wherein the address is an Internet
2 Protocol (IP) address.

1 Claim 17 (original): The method according to Claim 16, wherein the IP address is an IP version
2 4 address and wherein there are 4 components in each IP version 4 address and thus 4 arrays.

1 Claim 18 (original): The method according to Claim 16, wherein the IP address is an IP version
2 6 address and wherein there are 16 address components in each IP version 6 address and thus 16
3 arrays.

1 Claim 19 (currently amended): A method for providing fast and efficient address lookup for an
2 address comprised of a plurality of address components, the method comprising the steps of:
3 creating a plurality of arrays comprising an array for each of [[the]] a plurality of address
4 components which together comprise an address, wherein each array comprises a plurality of
5 entries which are indexed using values of the address component for which the array was created;
6 and

7 storing entries and information for each address to be subsequently looked up, further
8 comprising the steps of:

9 creating an entry for a particular address using the plurality of arrays by indexing
10 into a highest-order one of the arrays using a most-significant component of the particular
11 address as an index element; setting a flag associated with the index element to on if a bit mask
12 associated with the particular address indicates that the next-most-significant component of the
13 particular address is considered significant, and setting the flag to off otherwise; and repeating
14 the indexing and setting, using the next-highest-order one of the arrays and the next-most-
15 significant component of the particular address, while the bit mask indicates that the next-most-
16 significant component of the particular address is considered significant; and

17 storing information associated with the particular address [[value]] in a storage or
18 memory location associated with a last significant component of the entry, wherein the last
19 significant component is determined by [[a]] the bit mask associated with the particular address,
20 and setting a pointer field associated with the last significant component of the entry to point to
21 the storage or memory location.

1 Claim 20 (previously presented): The computer program product according to Claim 1, wherein
2 the computer-readable program code means for retrieving is performed by a plurality of distinct
3 processors, each operating on different ones of the components of the selected address value.

1 Claim 21 (previously presented): The computer program product according to Claim 1, wherein
2 the stored information for each of the addresses comprises routing table information associated

3 with a route to that address.

1 Claim 22 (previously presented): The system according to Claim 7, wherein the means for
2 retrieving is performed by a plurality of distinct processors, each operating on different ones of
3 the components of the selected address value.

1 Claim 23 (previously presented): The system according to Claim 7, wherein the stored
2 information for each of the addresses comprises routing table information associated with a route
3 to that address.

1 Claim 24 (previously presented): The method according to Claim 13, wherein the retrieving step
2 is performed by a plurality of distinct processors, each operating on different ones of the
3 components of the selected address value.

1 Claim 25 (previously presented): The method according to Claim 13, wherein the stored
2 information for each of the addresses comprises routing table information associated with a route
3 to that address.

1 Claim 26 (currently amended): The method according to Claim 19, further comprising the step
2 of subsequently looking up a selected address [[value]] by retrieving the stored information
3 associated with the selected address [[value]] from the plurality of arrays.

1 Claim 27 (new): The method according to Claim 26, wherein the retrieving step further
2 comprises the steps of:

3 indexing into the highest-order one of the arrays using the most-significant component of
4 the selected address as the index element;

5 while the flag associated with the index element is set on, indexing into the next-highest-
6 order one of the arrays using the next-most-significant component of the selected address as the
7 index element, thus reaching a selected one of the arrays;

8 if the pointer field associated with the selected one of the arrays points to a storage or
9 memory location containing stored information, using that stored information as the retrieved
10 information, and if not, iteratively checking the pointer field associated with a next-previous-
11 highest-order one of the arrays until the pointer field associated therewith points to a storage or
12 memory location containing stored information for using as the retrieved information or until no
13 more ones of the arrays remain for checking, in which case no stored information is retrieved.